REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*
21-09-2007	Conference Paper	01 Sep 2006-31 Aug 2007

4. TITLE AND SUBTITLE

Current Capabilities of Digital Beamforming

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
62204F

6. AUTHOR(S)

David D. Curtis, Daniel N. Spendley, Danh Q. Luu

5d. PROJECT NUMBER
4916

5e. TASK NUMBER
HA

5f. WORK UNIT NUMBER
01

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

AFRL/SNHA
80 Scott Drive
Hanscom AFB MA 01731
01731-2909

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Electromagnetics Technology Division Source Code 437890
Sensors Directorate
Air Force Research Laboratory
80 Scott Dr
Hanscom AFB MA 01731-2909

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL-SN-HS

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
AFRL-SN-HS-TP-2007-0011

12. DISTRIBUTION / AVAILABILITY STATEMENT
Approved For Public Release; Distribution Unlimited.

13. SUPPLEMENTARY NOTES
Presented at the 2007 Antenna Applications Symposium, Monticello, Illinois, 17-20 Sep 2007. Cleared for Public Release by ESC/PA: ESC 07-1055, dated 10 Sep 07. This is a work of the United States Government and is not subject to copyright protection in the United States.

14. ABSTRACT

This paper surveys the current state of the art of digital beamforming (DBF) with an emphasis on current capabilities and practical implementation techniques. Digital control of phased arrays in both receive and transmit modes will be covered, including such key issues as hardware, integration, processing load, and cost. The viability of future trends in DBF will be examined in terms of feasibility and cost of the enabling technologies of adaptive hardware and embedded software.

15. SUBJECT TERMS
Digital Beamforming, Antenna Technology, Phased Arrays

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			Richard T. Webster
Unclassified	Unclassified	Unclassified	UU	35	**19b. TELEPHONE NUMBER** *(include area code)* NA

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39.18

CURRENT CAPABILITIES OF DIGITAL BEAMFORMING

David D. Curtis, Daniel N. Spendley, Danh Q. Luu
Air Force Research Laboratory, Sensors Directorate
Electromagnetics Technology Division
Antenna Technology Branch - AFRL/SNHA
Hanscom AFB, MA 01731-2909

Abstract: This paper surveys the current state of the art of digital beamforming (DBF) with an emphasis on current capabilities and practical implementation techniques. Digital control of phased arrays in both receive and transmit modes will be covered, including such key issues as hardware, integration, processing load, and cost. The viability of future trends in DBF will be examined in terms of feasibility and cost of the enabling technologies of adaptive hardware and embedded software.

1. Introduction

Digital beamforming was first introduced in the 1980s [1]. Initial experiments validated the basic concepts of DBF [2, 3] and demonstrated a wide range of adaptive null forming and direction finding algorithms that had been first developed for analog phased array antennas [4, 5, 6]. Following the initial interest in DBF, many specific details were then examined and several experimental test beds followed. Some of these prototyped real-time sampled data processing [6, 7], elimination of element pattern effects [8], various channel calibration approaches [9], the use of DBF for control of exotic conformal arrays, and experimental prototypes at nearly all the common radar frequencies [10, 11, 12]. While much has been demonstrated, most of this work remained in the realm of the laboratory, and very little of it was at a technology readiness level suitable for transition to either the commercial sector or to military applications. That was due to an initially slow evolution in the state of the art in digital array channel hardware.

Over the past four to five years, however, there has been a virtual explosion of high performance commercial-off-the-shelf (COTS) digital array channel hardware and signal processing chips. Leading the way are tremendous advances in field programmable gate array (FPGA) logic chips. FPGAs are now approaching the speeds of application specific integrated circuits (ASIC) and other dedicated digital signal processing (DSP) chips, yet the ease of programming embedded logic in FPGAs permits the array designer to make significantly simpler redesigns in order to scale the array or achieve different functional goals. ASICs also enable great levels of digital signal processing performance, and they may be the best choice for hardware designs which do not change frequently, but usually, ASIC design is not a trivial task. It often involves an iterative process of partial redesign

and re-layout in order to achieve desired functionality and bring signal timing issues within specified criteria. DSPs offer speed and programmability in one package. These forms of processors are all integrated circuits. In some applications where size, weight, and power consumption (SWaP) are not at a premium, high speed computer systems with real-time operating systems are sufficient. For airborne applications where SWaP and processing speed is truly important, the FPGAs, ASICs and DSPs win hands down.

The analog to digital converter (ADC) has long been the single most important component in a digital array. It not only enabled digital beamforming, it also limits array performance to a certain number of bits of digital resolution at a particular maximum sampling rate. As the sampling rate goes up, the effective number of bits comes down, and visa versa. These two parameters are very tightly coupled and the state of the art in ADCs advances at a rate of about 1 bit of added resolution every 4 years or so, for a specific sampling frequency [13]. However, the recent evolution of processing hardware has resulted in companies marketing the latest and very best available A/Ds, FPGAs, and random access memory (RAM) chips on demonstration boards that have standard personal computer interfaces like peripheral component interconnect (PCI) and virtual measurement environment (VME). The available performance has rocketed over the last two to three years, and prices are starting to come down. This has allowed array designers to prototype digital channel hardware fairly easily. In turn, digital array channel hardware is now in demand more than ever in many array antenna applications, and in certain cases, can be fielded as a subsystem in an phased array antenna system with high confidence.

This paper aims to provide an update on the current state of the art of digital beam-forming technology. We begin by discussing a generalized digital array architecture which can be morphed to suit commercial radar and satellite communications applications and a wide variety of military missions, including radar, communications, signals intelligence, and satellite ground station telemetry, tracking and commanding, to name just a few. Changes in mission cause changes in key performance parameters such as scan volume, center frequency, instantaneous bandwidth, dynamic range, the number of simultaneous beams, and data processing load. In the next two sections of this paper we describe the functionality and the key components of receive array channel hardware and transmit array channel hardware, respectively, including elementary forms of digital beamforming on receive and transmit, and we discuss how the basic channel hardware changes with changes to the mission key performance parameters. Next, we describe processing for DBF, including algorithms, how channel weights are formed, the hardware that is used, data processing workload, or simply, processing load, and the impact on processing due to changes in dynamic range and bandwidth. Finally, we address future trends in DBF, including modularity, multi-functionality, nonlinear digital filtering, receiver-on-a-chip (ROC) technology, wideband null forming, localized signal generation, and orthogonal waveforms.

2. Digital Array Architectures

Digital beamforming is the term often used to refer to the merger of three interrelated technologies: digital transceivers, digital signal processing, and phased array antennas. Digital array architectures are the desired end results of merging these technologies. A block diagram of a generalized N-channel digital array architecture is shown in Figure 1.

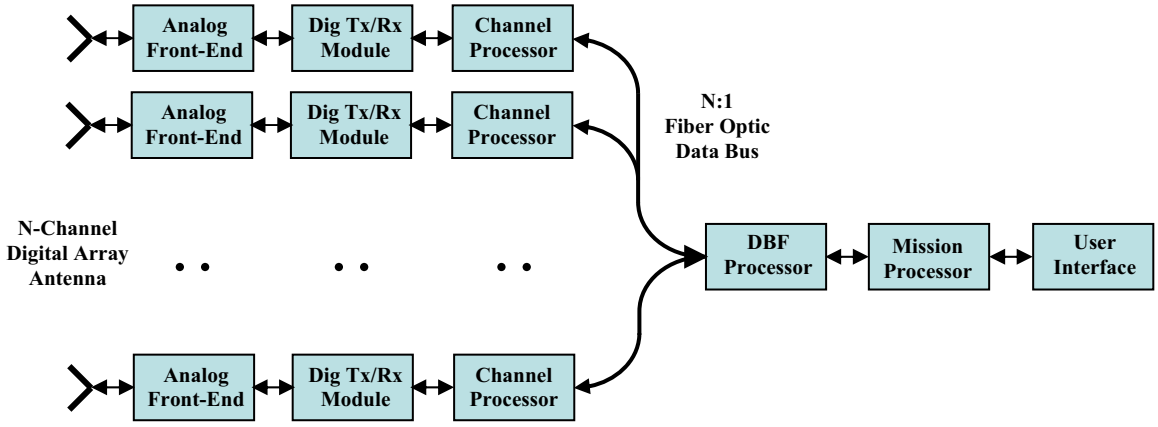


Figure 1: Block diagram of a generalized N-channel digital array architecture.

This generalized architecture is composed of N parallel digital array channels, an N:1 data and command signal distribution network, shown here as a fiber optic bus, and dedicated processors that perform digital beamforming computations and mission specific signal processing tasks. A user interface also may be included, but is not required in general. Depending on cost and the suitability of available hardware to meet mission requirements, each digital array channel may be connected to an individual array element, an array column or row, or a subarray. An analog beamformer is required to combine individual array antenna elements into columns, rows, or subarrays before they are connected to the digital array channel.

All the digital signal processing functions may be handled by a centralized processor, as was typically the case in the early experimental DBF prototypes. However, the current approach is to distribute the processing load across the array in N channel processors, as well as perpendicular to the array in a DBF processor and a mission-specific processor. Distributed processors reduce the overall data throughput in the architecture because each time two data streams are added together, the result is a single data stream representing their sum. While fairly generalized, the digital array architecture shown in Figure 1 emphasizes this approach.

2.1 Analog Front-End

Each digital array channel is composed of an analog front-end, a digital transceiver module and a channel processor. The analog front-end is the hardware interface between the array antenna and the digital electronics. It typically contains microwave frequency low noise amplifiers, power amplifiers, mixers, and switches. A block diagram of a typical analog front-end is shown below in Figure 2.

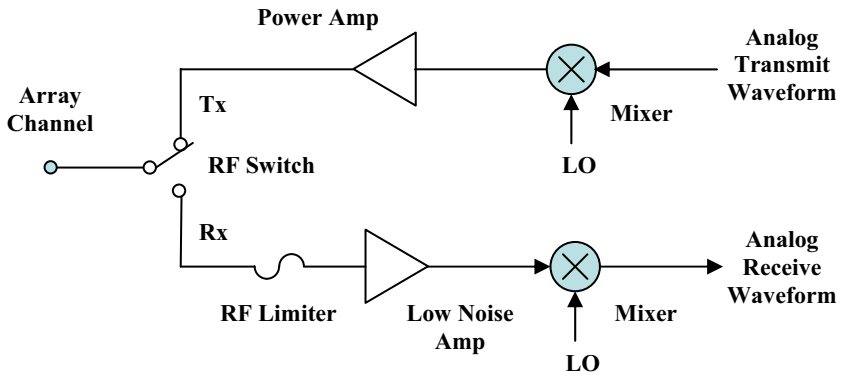


Figure 2: A typical Analog Front-End module for digital arrays.

In this diagram, the array channel is connected to either the transmit path or receive path of the analog front-end by a switch. On the transmit side of the module, the analog transmit waveform is up-converted by a mixer, amplified, and routed to the array channel through the switch. On the receive side, the signal from the array channel is routed by the switch into a limiter that protects the sensitive receiver electronics from high power signals, a low noise amplifier boosts the signal, and a mixer performs down-conversion. The resulting analog receive waveform is passed to the digital transceiver module. For full-duplex transmit and receive operation the switch may be replaced with a circulator. For frequencies around 500 MHz and lower, the mixer hardware and LO signals may be eliminated. Two or three stages of mixers and LO signals may be required for GHz frequencies.

2.2 Digital Transceiver Module

The second element in the digital array channel hardware chain is the digital transceiver module, shown in Figure 3. Detailed block diagrams of the digital transceiver module and the channel processor are shown with data paths represented by dashed lines and control signals represented by solid lines. The digital transceiver module is composed of four functional blocks. Each functional block performs a variety of related functions, as enumerated within the blocks in the figure.

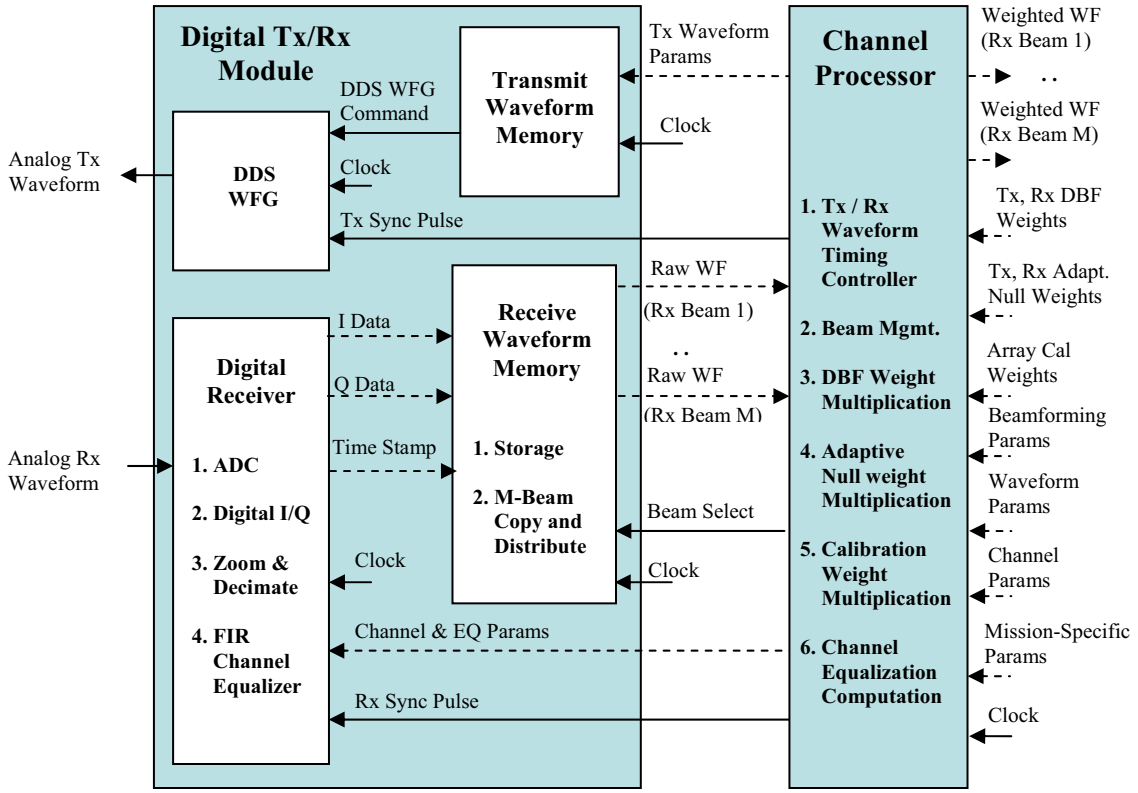


Figure 3: Block diagrams of the digital transceiver and channel processor modules.

The digital transceiver module transmit path is composed of a transmit waveform memory and a direct digital synthesis waveform generator (DDS WFG). In this scheme, a set of transmit waveform parameters are inputs to the transmit waveform memory. These parameters may include instantaneous bandwidth, center frequency, amplitude, digital modulation format, duty cycle, etc. These parameters are translated into command words for the DDS WFG. All modules in the digital array architecture run on the same clock signal. When a transmit synchronization pulse arrives at the DDS WFG, it triggers the DDS WFG to generate and send the desired analog transmit waveform to the analog front-end for up-conversion.

The receive path is composed of a digital receiver and a receive waveform memory. Channel configuration parameters are sent to the digital receiver as control variables. They set the sampling rate, instantaneous bandwidth, and digital resolution. Channel instantaneous bandwidth is controlled by zoom and decimation filtering, and a channel equalization filter is set up by the equalization parameters. The receive synchronization pulse triggers the digital receiver to sample the incident analog received waveform on the next clock pulse. The digital receiver decomposes the raw sampled data into in-phase (I) and quadrature (Q) data streams. This data, along with a time stamp, is briefly stored in

the receive waveform memory. If M multiple beams are desired, the received data is copied and distributed into M identical raw data streams. The receive waveform memory uses the beam select signal to select data corresponding to one or more of the M beams which will be forwarded to the channel processor.

2.3 Channel Processor

The third and final element in the digital array channel is the channel processor, which is also shown in Figure 3. This is the first of the processors. It accepts a wide range of parameters that are generated in the mission-specific processor, as well as a variety of complex-valued data weights generated by the DBF processor. Some of these parameters are passed through to the digital transceiver module, and others are used in the channel processor to perform specific array signal processing functions. This includes application of array calibration weights to the raw I/Q channel data, which correct for differences in amplitude and phase between any and all pairs of array channels. Once the array is calibrated, the beamforming weights can be applied to the calibrated channel data along with the null forming weights. Nulls may be formed deterministically, but most often, this is done adaptively, to counteract strong interfering signals. Alternatively, the array calibration weights, beamforming weights and adaptive null weights may be lumped into a single composite weight for each array channel. The channel processor performs a complex-valued multiplication of the raw I/Q data stream with the weights. A beam management function in the channel processor generates the beam select command that is sent to the digital transceiver module in order to extract the stored raw I/Q waveform data from the receive waveform memory. The selected waveform data streams are weighted in the channel processor and passed to the DBF processor where they are summed to form beams and nulls, and for use in other array processing such as direction finding.

2.4 N:1 Fiber Optic Data and Command Bus

The weighted data streams from all N channels must be added to form beams and nulls. This can be done in a systolic processor, which has several processor nodes distributed across the array backplane. Each node performs a mathematical operation on its input data, and passes its output to the next processor node. One node is the final processor which completes the mathematical operation. These architectures are typical in DBF systems that support a single dedicated application, where a variety of beamforming and nulling algorithms is not required. When flexibility is needed, a centralized processor often is the answer. In Figure 4, we show an N:1 fiber optic data bus that performs the collection and distribution function between the N digital array channels and the DBF processor. This bus has electro-optic data transceivers at the end of each optical fiber, and it contains fiber optic power dividers that multiplex the data onto a single fiber that is incident to the DBF processor. Parameters, weights and the clock signal flow from the DBF processor to the channel processors. Array pre-calibration channel data and weighted waveforms from the N receive beams flow back.

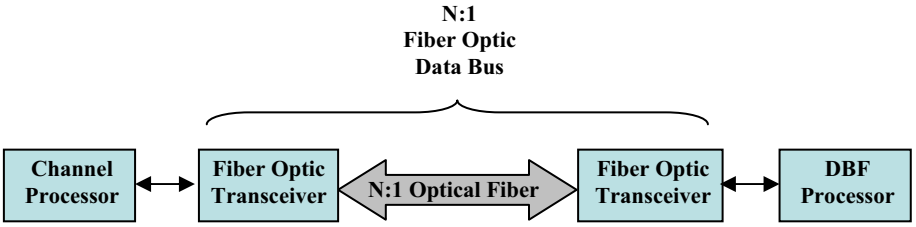


Figure 4: The N:1 Fiber optic data bus manages the transfer of data and command signals between the channel processors and DBF processor.

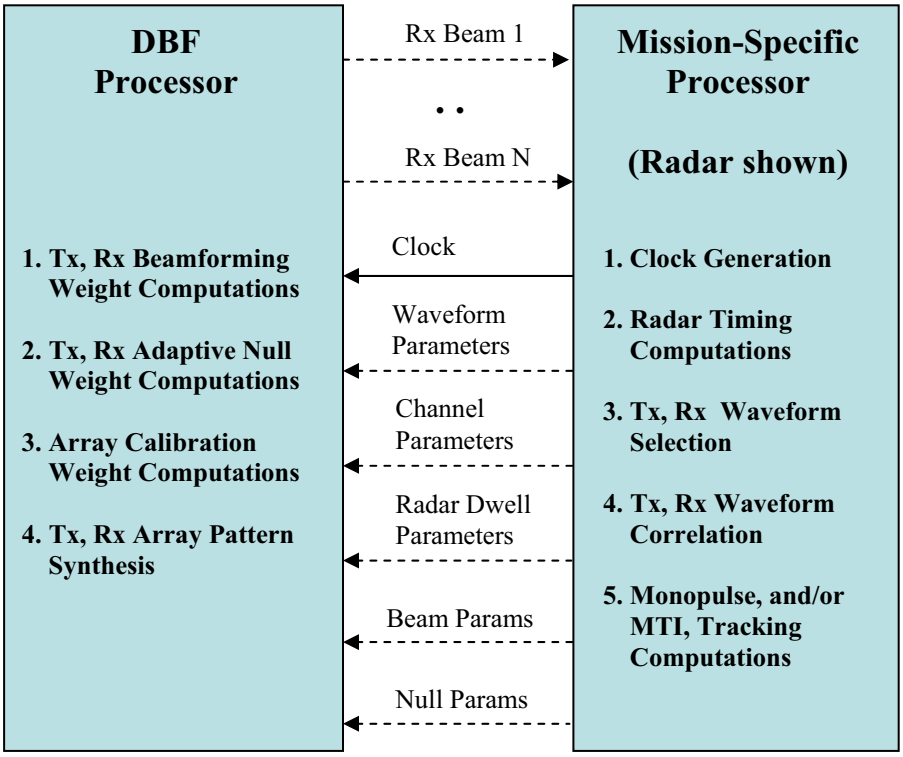


Figure 5: Illustration of the computational roles of the DBF processor and the mission-specific processor, which in this figure, is shown as a generalized radar signal processor.

2.5 DBF Processor

The DBF Processor performs four main functions. First, it is the collection point for the channel data. Channel data is accumulated in a covariance matrix which represents the statistics of the environment in which the array antenna is operating. Second, the DBF

Processor inverts the covariance matrix, and forms the weights according to an adaptive algorithm. Typically, the weights that result from adaptive beamforming algorithms are composite weights that include the beamforming and nulling behavior. The DBF processor sends the computed weights back to the channels where the channel processors apply the weights to the raw data streams, and third, the returning weighted channel data streams are summed in the DBF processor, thus forming the beams. This beam data is then passed on to the mission-specific processor. While adaptive algorithms are designed to converge to a solution to a particular parameter minimization or maximization problem, some of the more advanced digital beamforming algorithms can be classified as array pattern synthesis. The method of alternating projections is one such algorithm. It allows the array designer, or potentially, the mission-specific processor to generate a spatial mask that represents the desired array far field power pattern. The algorithms then adapts until a set of weights are found which closely approximate the mask [14]. Our team has used this algorithm to obtain beamforming weights for an experimental conformal digital array that wrapped completely around an aircraft wing. Such an array would normally have been an impossible application for most adaptive beamforming algorithms due to the complexity of the conformal array curvature [15].

The fourth main function of the DBF processor is calibration. To perform calibration the DBF processor shuts down all normal array operation, and sends unitary-valued weights to the channel processors so that the returning weighted channel data streams contain the only the raw I/Q data from each channel. The DBF processor selects one channel as the reference channel, and data from all the other channels is normalized to the reference channel data. The complex-valued calibration weights are obtained by the normalization process. Once this has been done, the DBF processor allows the array to resume normal beamforming operations and the calibration weights are combined with the beamforming and nulling weights to form the composite weights for each channel.

2.6 Mission-Specific Processor

The mission-specific processor generates the mission-specific parameters, the waveform parameters, the system clock signal, and it controls the timing between transmit and receive beams. In Figure 5, the mission-specific processor is shown as a generic radar processor. Different parameters would be generated if this digital array architecture was to be used in a communication system or an intelligence collection system. Based on these mission parameters, up to N transmit beams are formed, or up to N receive beams are formed and returned from the DBF processor to the mission-specific processor.

3. Key Components of Receive Architectures

In the previous section we described a generalized digital array architecture that emphasized distributed processing as a means of reducing data volume, and ultimately, processing load. In this section we explore variations of the digital receiver hardware, shown in Figure 3, and the overall receive mode digital array architecture.

The main components of a digital beamforming system's receive architecture are shown below in Figure 6. The functionality of each individual block may vary greatly between different DBF systems, and each is heavily influenced by its neighboring components. For instance, the techniques and algorithms used to perform beam weighting are extremely dependent upon the sampled data's format. The overall system functionalities will also help determine the detailed architecture by specifying the operational RF spectrum, instantaneous and system bandwidths, timing constraints, and other factors key to any DBF system's mission. Moving through and considering each operational block in step-by-step fashion will allow a full understanding of the hardware's capabilities and limitations as well as options for implementing efficient and practical DBF systems.

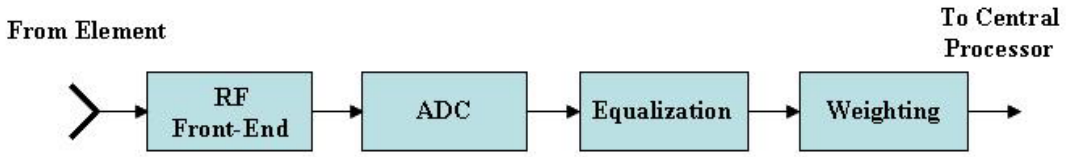


Figure 6: Illustration of the key processes in a typical DBF receive channel. Some architectures may combine one or more blocks, but these processes are still performed.

3.1 Analog Front-End

The radio frequency (RF), or analog front-end demodulates the message signal from the RF carrier to an intermediate frequency (IF), or to baseband, where sampling and signal processing are performed. The input bandwidth and output bandwidth are important design parameters that govern analog front-end design. Designing a wideband tunable front-end used for surveillance is markedly more complex than designing a narrowband front-end for a fixed frequency communications system. The wideband front-end must span the tunable bandwidth, and hold the noise floor down to preserve spur-free dynamic range. This often requires expensive custom components and considerable design expertise. If the tuning range requirements exceed the bandwidth limits of state of the art ADCs, one approach is to parse the tuning range into smaller frequency ranges, or analog sub-bands, that can be handled using available components. Diplexers, triplexers, and active circuit versions of such hardware are typically employed for this purpose. If the instantaneous bandwidth is too wide for the available components, the same approach may be used. By comparison, the design of a narrowband front-end is trivial unless it requires an extremely high spur-free dynamic range that exceeds state of the art ADCs. In this case, custom multi-stage ADCs may be constructed to sample the full dynamic range, but this is not frequently done due to the complexity and cost of the custom integrated circuit design. In addition to a relatively simple design process, COTS narrowband front ends are readily available from a large number of vendors.

Differences between narrowband and wideband front-ends often can be obvious if one examines their implementation. The relative ease of designing and implementing narrowband components and transmission lines enables them to be implemented in compact, power efficient packages such as the 2" x 2" printed circuit board shown in Figure 7. Issues with wideband component manufacturing and impedance matching often makes state of the art wideband front-end specifications harder to achieve.

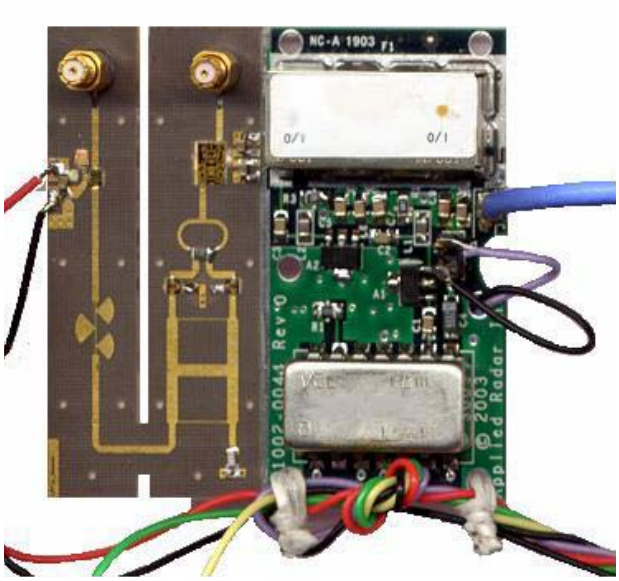


Figure 7: Example of a custom X-band analog front-end with a 15 MHz instantaneous bandwidth, that utilizes a signal stage mixer to down-convert from 10 GHz to 75 MHz.

3.2 Analog to Digital Converter

The next component in the receive channel block diagram in Figure 6 is the analog to digital converter. ADCs with sampling speeds ranging from a few kilohertz to multi-gigahertz are currently available from a limited number of commercial retailers. When combined with a properly designed analog front-end, ADCs can provide more than a gigahertz of instantaneous bandwidth. The maximum sampling rate and the digital word length, or number of bits representing each sample, are two important parameters to consider when selecting an ADC for a particular digital array application. The digital word length governs the sampled signal's potential resolution and helps establish both the dynamic range (DR) and signal-to-noise ratio (SNR) of the receive channel hardware. It can be shown that 1 bit of digital resolution corresponds to 6.02 dB of analog dynamic range [16]. Thus, a 48 dB dynamic range may be sampled using an ADC that has an effective number of bits (ENOB) of at least 8 bits. The DR and SNR of the total system must be calculated by considering the contributions of all the components in the receive channel and processing chain. However, the DR and SNR of the ADC alone provides a

good initial estimate of the system's performance limits. The digital word length and the maximum sampling rate also establishes the data throughput rate which the rest of the signal processing components must accommodate. The data throughput rate of a digital array consisting of a small number of narrowband channels may seem very low, but the data throughput rate becomes a limiting factor in a large scale digital array containing hundreds or thousands of elements. Systems such as these could easily drive the data throughput rate into the terabytes per second range, causing the data throughput rate to be a limiting factor in a practical system realization. Design trades must be performed to determine the most cost effective and high performance design for a given application.

The analog front-end and the ADC have the greatest effect on the digital receiver design compared to any other combination of the components in the receive channel. Consider a narrowband digital receiver operating in the ultra-high-frequency (UHF) band. The low center frequency and narrow instantaneous bandwidth in this case enables the designer to chose how the sampling will be performed. As one option, a low cost ADC having a low maximum sampling rate may be used if a suitable analog front-end is designed to down-convert the UHF signal to baseband. As a second option, down-conversion may be eliminated altogether if a more expensive high speed ADC is used which can sample the UHF signal directly. The design trade space is usually a multi-dimensional problem. In this simple example, it is a matter of the cost, size, weight, and power consumption of the components taken in context with the performance of each approach, including dynamic range, noise figure, and signal-to-noise ratio.

3.3 Channel Equalization and Array Calibration

A simple narrowband normalization approach to array calibration was briefly described in section 2.5 for the generalized digital array architecture. However, a variety of other approaches may be used to ensure that all channels in the array are as highly correlated as possible. An array is said to be calibrated when all its channels have nearly identical insertion losses and path lengths. In a digital array this means the channels are nearly identical mathematically, usually to within the least significant bit of the processor and data distribution hardware. A calibrated array is capable of forming deep nulls in its far field power pattern because of the high degree of correlation between the channels. However, when the array is of moderate to wide bandwidth the analog front-end components typically have frequency responses that are not well correlated from one channel to the next, and all of this precision is lost quite quickly. In order to calibrate arrays of moderate to wide bandwidths, channel equalization must be used in conjunction with array calibration.

The goal of channel equalization is to flatten the magnitude and linearize the phase of the channel frequency response over the entire instantaneous bandwidth of the channel. This is accomplished in a two-step process. First, the channel is fed with a linearly swept sinusoidal signal and the frequency response is measured by taking samples of the output at a finite number of discrete frequencies. That response may be stored in memory, but

ultimately is used to form an inverse filter whose frequency response is the complex conjugate of the measured frequency response of the channel. The second step is the design of a finite impulse response (FIR) filter [17, 18], which is often implemented as a digital tapped delay line with coefficients related to the frequency response of the inverse filter. The convolution of the channel response with the inverse filter produces the desired result. Because the channels in the array may have different frequency responses, it is necessary to equalize each channel independently, but this may be done in parallel if the processor architecture can support it. Once each channel is equalized, the narrowband normalization approach [19] may be used to perform array calibration. This is because the channel frequency responses now have constant magnitudes and linear phases within each channel, but the magnitudes and phases differ from one channel to the next.

Alternatively, one may calibrate the array and achieve a form of channel equalization over the instantaneous bandwidth using a different technique. After measuring and recording the channel frequency responses of all the channels, the frequency response magnitude and phase at the k^{th} discrete frequency may be compared across all N channels in the array. This comparison may be a normalization to the amplitude and phase of one channel chosen to be the reference channel, or it may be the average complex value of all the channels at this one discrete frequency. This is narrowband calibration which is valid only at the k^{th} discrete frequency. If this process is repeated at all K discrete frequencies of the measured channel frequency responses, K discrete frequency array calibrations will be performed and each channel in the array will have K calibration coefficients which span the entire instantaneous bandwidth. The channels will not be equalized in a true sense, meaning that the frequency response magnitudes will not be flat and the phases will not be linear, but the relative differences in magnitude and phase between all the N channels will be nearly zero at each discrete frequency.

Like calibration, the equalization process typically requires the array to cease all normal operations while the channel frequency responses are measured and the coefficients of the inverse filters are calculated. However, in some applications, it may be necessary to perform channel equalization and array calibration more frequently. This may be necessary when the array undergoes some physical change such as the displacement of an element due to mechanical deformation, vibration or thermal expansion and contraction. Situations such as these require a re-evaluation of the relative spatial dimensions between each receive element before further equalization can occur.

3.4 Local Memory Blocks

Physical memory blocks may be inserted in the receive array hardware to support calibration and equalization, but these can also support data transfers, further processing, and data re-evaluation. Due to its wide range of uses, memory can be integrated into or around many of the functional blocks in the receive architecture or included separately as can be seen in Figure 8. A small amount of memory connected to the ADC enables the processor to collect raw data for calculating adaptive beamforming weights or playback

of received incoming waveforms. Large blocks of memory can also be attached to the channel or array processing board, allowing for a vast amount a data to be collected before being sent to a central processor for final calculations and evaluations. These large data depositories are able to collect data at a high rate, allowing the DBF system to run at its maximum speed for a limited time and then transfer the recorded data to the processor at speeds realizable using current data bus technology. In many applications, running in this "quasi-real time" mode is sufficient for operation given that relatively large wait times may be necessary in between processing periods.

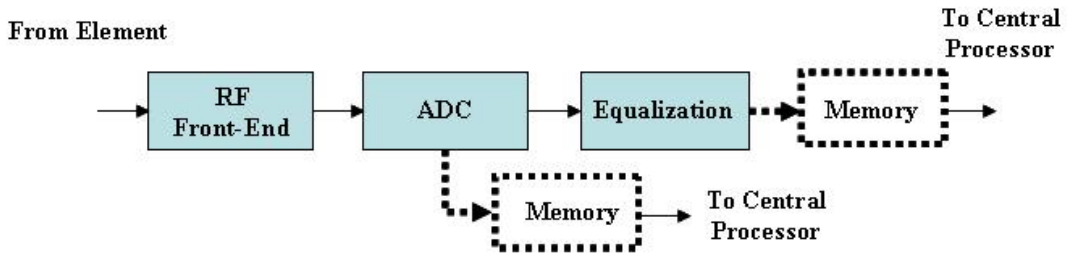


Figure 8: Examples of optional memory locations for DBF receive architectures. On-board memory can be inserted almost anywhere on the receiver board permitting the available real estate exists. The above architecture includes optional memory after the ADC as well as after equalization for weight generation and beamforming offline.

3.5 Weighting

The last block in the general receiver architecture in Figure 6, applies beamforming weights and other signal processing algorithms to the sampled data. Beamforming is often thought to occur in a central processing unit, but application of weights may be performed at the channel level. This allows a designer to apply the beam weights at either the channel or the central processor. In the narrowband case, the beamforming weights can simply be represented as a complex number representing an amplification and phase shift. Most array processing textbooks [20, 21], as well as the current beamforming literature [22], represent the weighting scheme as the inner product between two complex vectors, the input signal, $\boldsymbol{x}$, and the corresponding weights, $\boldsymbol{w}$.

$$y = w^{*} x \tag{1}$$

Equation 1 provides the mathematical relationship for the weight-and-sum beamforming architecture familiar to both the analog and digital array communities. The inner product calculates the power of the received signal in a direction specified by beam weights, $\boldsymbol{w}$.

In general, three main architectures exist for applying beamforming weights:

1. ***Real-time***: All weighting is performed at the same rate data is inputted to the digital processor. The resulting data is continuously streamed out of the processor for further analysis without interruptions.

2. ***Quasi-Real-time***: Weighting may be performed in a real-time manner, but a data bottle neck occurs at some point in the architecture resulting in a time lag between available processed data sets. Bottle necks are often caused by a large data throughput which the digital bus hardware is unable to accommodate.

3. ***Offline***: Sometimes referred to as "Poor Man's DBF". All channel data is placed in a data depository to be processed at a later time in software. This architecture allows digital beamforming to be implemented at a basic, cost efficient level, but not an operational ready level.

Real-time and quasi-real-time architectures are most often implemented using components designed for parallel processing, such as FPGAs and ASICs. These chips provide a DBF designer with an abundance of resources including a flexible application platform, high number of input/output (I/O) pins and specialized processing circuitry. Of these two options, FPGAs provide the most design flexibility because of they can be reconfigured easily. This offers the designer a large number of debugging options as well as the ability to add additional processing capabilities for applications defined at some time after the initial design. The ASIC provides a more restrictive processing block because its architecture is hardwired, and it is inherently non-adaptable to future applications. But since the processing components are predefined, ASICs typically require less power and may achieve lower costs when manufactured in high numbers compared to the FPGA.

4. Key Components of Transmit Architectures

A general block diagram highlighting the main components of the DBF transmitting architecture is shown below in Figure 9. Transmit beamforming is often considered the "dual" problem to receive beamforming and thus a simple reversal of the receive chain components is expected to achieve a high performance system. This general philosophy is not without merit, but is still a somewhat naive design process. As with the receive architecture, the operation of every functional block affects the other components in the transmitting chain, but a new, important design issue arises that is not an issue for DBF on receive. That is, the coherence of the channels relative phases. The channel-to-channel signal coherence will be shown to be both the transmitter's enabling feature as well as its inhibiter. For this reason, each of the transmitter architecture's components will be discussed from the right to left to provide a greater understanding of the signal coherence's importance from the steps of signal generation to transmittance.

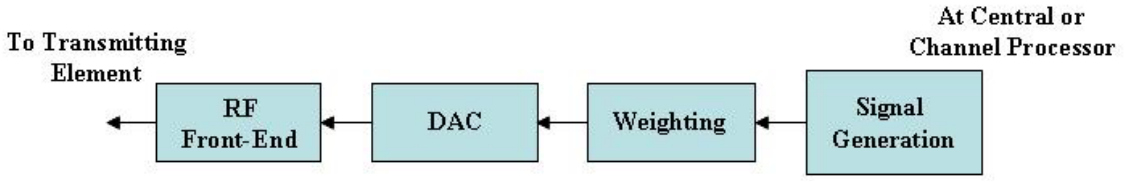


Figure 9: Key components of general DBF transmit architecture. The message signal is first generated at the far right and is processed through the functional blocks for weighting and RF up-conversion by moving left.

As in the receive case, the transmitting beamformer must implement a desired phase progression across the transmitting channels in order to properly cohere the individual signals for beamforming. Similar to a basic, general receive architecture where many designs can be implemented to receive, sample, and apply weights to the incoming signal, multiple options exist for implementing phase progressions across the transmit channels. One such option implements a single desired digital signal at a central processor and distributes it to each channel where weights are applied in parallel, as illustrated in Figure 10, below. Although this approach reduces the quantity of signal generation hardware required, it requires a large volume of data to be transferred from the central processor to the channel processors, which ties up valuable power and data bus resources.

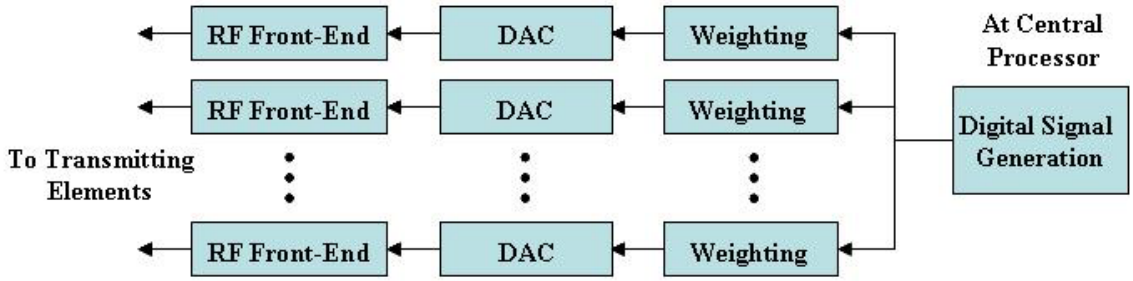


Figure 10: Architecture to implement transmit phase progressions from a single signal generator located at the central processor. Signal is then passed to channels processors which perform both the weighting and up-conversion.

Another architecture option utilizes waveform generation control signals at the channel level as seen in Figure 11. The resulting system takes advantage of current signal generation techniques, but requires the control and waveform timing to be extremely accurate. In both architectures, it is pertinent that the channel responses are taken into account during the initial signal generation. Failure to consider the different responses will result in a corrupted phase progression at the transmitting array.

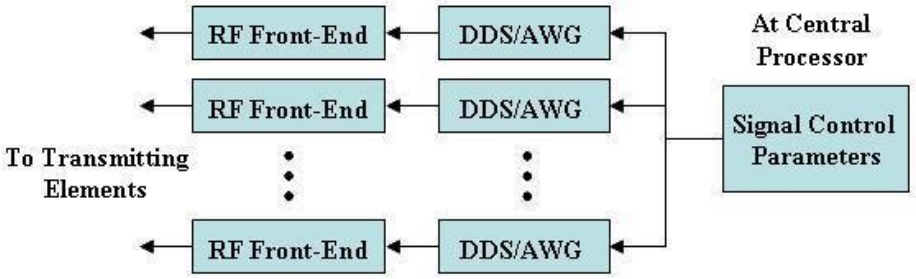


Figure 11: Architecture to implement transmit phase progressions from multiple signal generators located on each channel processor. Control and timing parameters for waveform generation are passed to the channel processor to generate the signal. Signal phasing could be performed using weights or through signal's control parameters.

The memory blocks in a transmit DBF architecture can be used to alleviate some of the strict timing constraints mentioned in the previous paragraph. By utilizing memory to hold actual digital waveforms or their specific generating parameters such as frequency, amplitude and phase, implementation of the resulting waveforms is now focused on the control timing rather than the waveforms themselves. Waveform parameters stored in memory can be transferred easily to signal generators such as Direct-Digital-Synthesis (DDS) chips and actual digital waveforms can be readily applied to an Arbitrary-Waveform-Generator (AWG). These components are key building blocks in transmit DBF systems. These possible enhancements demonstrate that using memory blocks for waveform generation may increase the control signal overhead, but decrease the transmit system's total data throughput, thus simplifying the overall design.

As has been stressed throughout this section, that accurate signal generation is a key component of a transmit DBF system. Two of the most popular techniques for signal generation leverage the abilities of DDS and AWG chips. DDS chips rely on accurate timing circuitry and phase accumulators to produce digital waveforms [23]. Common features on these chips include frequency, phase, or amplitude modulation modes which are specified by input parameters made available either at external I/O ports or within their on-chip memory components. This enables accurate, simple waveform generation, provided that all timing constraints are met. In addition to pre-programmed modes, AWG chips possess the capability to transform previously stored digital waveforms into their analog counterparts. This allows for the user to create and upload waveforms to the channel processors for signal generation. Both the DDS and AWG chips heavily rely on digital-to-analog chip (DAC) technology which is often integrated into the packaging to produce the output analog waveforms. DAC technology currently allows the creation of waveforms exceeding 1 GHz of bandwidth, providing the transmitter with a great deal of flexibility for a variety of applications.

Even though the analog front-end design of a DBF transmitter is similar to a standard RF up-conversion chain, these components often cause the most headaches in designing a transmit channel due to the difficulty of eliminating channel-to-channel inconsistencies. Calibration on receive, as mentioned in Section 3.0, has been found to be a relatively simple processes since channel response measurements are readily available. Due to the nature of transmit DBF, more effort must be placed on accurate measurements of the transmit channel performance. Since channel response of the front end is highly sensitive to frequency, systems operating in different bands may encounter this issue with varying severities. For high frequency, stand-alone applications, the up-conversion process may require multiple RF stages requiring precise coherent local oscillator (LO) generation between channels. Applications at lower RF frequencies are achievable with current DAC technology and can avoid these issues, but nonetheless, the remaining analog components must be carefully characterized to implement an accurate phase progression.

5. Processing Algorithms and Hardware

In an array, a beam can be formed by delaying and summing the signal from each element. One of the key steps in beamforming is to accurately delay the signals at each channel so that they sum in phase. In a narrowband analog array, phase shifters can be used for beamforming instead of the time delay units. The received signal in each of the channels of the array consists of the message signal modulated upon the carrier. Small arrays with moderate bandwidth are classified as "narrow band" arrays because the delay of the message signal from channel to channel is insignificant when compared to the period of the message signal, and thus, can be ignored. The beamforming problem is then reduced to the problem of removing the time delay from the carrier signal, which can be accomplished by applying the appropriate phase shift to each channel. To form a beam in large arrays with moderate to wideband instantaneous bandwidths, a time delay is required to correct for the propagation delay from the wavefront that impinges on an array at various angles to the array aperture.

5.1 Wideband Beamforming via Fractional Sampling Delay

One technique to form a wideband beam is by using fractional sampling delay filtering. After the analog signal is digitized at a sampling period T, the digital signal can be delayed exactly, if and only if the delay value is an integer multiple of T. However, the delay required to form a beam at a specified direction may be only a fraction of T. One can round the delay amount to the nearest multiple of T but in cases where the delay amount is small, rounding adds significant delay error to the total amount of delay, which causes the array pattern sidelobes to rise.

Given a discretized signal x[n], if the desired time delay values D_a is a multiple of the signal sampling period T, then a delayed version of x[n] can be obtained by simply shifting x[n] by D samples. However, it is not immediately clear what x[n – D] should be if D_a is not a multiple of T or if D_a is less than T.

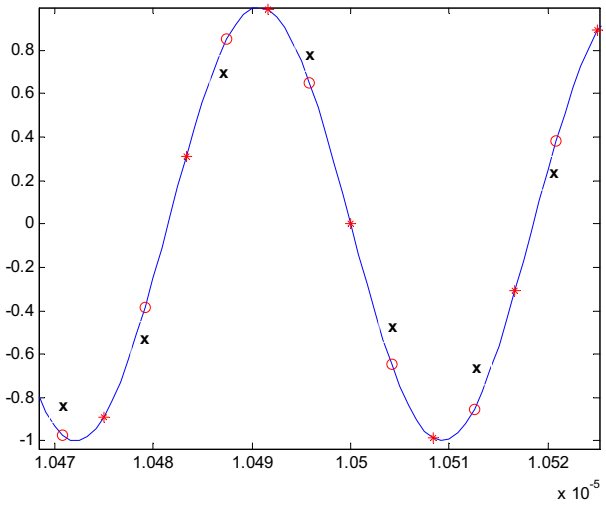


Figure 12: Time plots of a continuous signal x(t) are shown in solid blue and a discretized version of x(t), x[n], is shown in '*' red. Two possible versions for x[n – 0.5] are also shown as 'o' red and 'x' black.

For example, consider Figure 12 above, which shows time plots of a continuous signal x(t), its discretized version x[n] and two possible versions of x[n – 0.5], where 0.5 denotes a delay amount of half of the sampling period T. At first, from looking at the discrete samples x[n] alone there seems to be an infinite number of possibilities for x[n – 0.5] since we don't know the values that x[n] assumes between sample intervals. However if the sampling rate meets or exceeds the Nyquist requirement then there is a unique continuous curve x(t) that passes through x[n]. This means that x(n-D) is unique for any D. An obvious strategy to compute x[n – D] is to first reconstruct x(t) from x[n] using the Nyquist theorem [24],

$$x(t) = \sum_{m=-\infty}^{\infty} x[m]p(t - mT), \tag{2}$$

where T is the sampling period and p(t) is the ideal reconstruction filter, and

$$p(t) = \frac{\sin\left(\frac{\pi t}{T}\right)}{\frac{\pi t}{T}},$$

After x(t) is known, $x(t - D_a)$ can be computed by using (2), which gives the following.

$$x(t - D_a) = \sum_{m=-\infty}^{\infty} x[m]p[t - mT - D_a], \tag{3}$$

where D_a is a delay amount, in seconds. Finally, the desired delayed signal x[n – D] is simply a sampled version of (3),

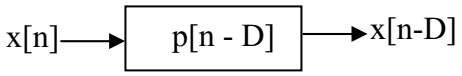
$$x[n - D] = \sum_{m=-\infty}^{\infty} x[m]p[n - m - D], \tag{4}$$

where p[n – D] is the ideal fractional delay filter, described as

$$p[n - D] = \frac{\sin((n - D)\pi)}{(n - D)\pi}, \qquad \text{for n} = -\infty,\ldots,-1,0,1,\ldots,\infty$$

.

Another way to express (4) is with the following:



One of the properties of the ideal infinite length fractional delay filter is that it delays the discrete signal x[n] exactly DT seconds for any fractional value D, but it does not distort x[n]. In practice, one must truncate the ideal fractional delay filter to some finite length so that the filter can be implemented in hardware. A simple way to truncate the ideal fractional delay filter is to multiply it with a rectangular window, which is defined to have a value of 1 between over time period and zero elsewhere. Because the rectangular window has very high sidelobes, the resulting windowed delay filter will have significant amplitude ripples in the frequency response especially at the edge of the band. One way to reduce the amplitude ripples and delay error of the truncated fractional delay filter is to reduce the sidelobe levels of the window [25]. For example, using a Chebyshev window with low sidelobes in place of the rectangular window causes the amplitude ripples of the truncated delay filter to be reduced significantly.

The group delay plot of a delay filter designed to have a delay value of 0.4 over about half of the normalized Nyquist bandwidth is shown in Figure 13. The integer delay amount can be neglected since one can compensate for it by shifting samples forward or backward in time. Even with only eleven taps and working with Matlab double precision, this delay filter had a maximum delay error of only 0.0004 of the sampling period. In practice when the filter coefficients are quantized to be implemented in DSP hardware, the delay error increases moderately. For a given filter length, there is an

inverse relationship between the usable bandwidth region and the peak error of the delay filter. An effective way to reduce the peak error of the windowed delay filter is to somehow reduce the usable bandwidth region of the filter. When the usable bandwidth region is reduced, the zeros of the filter are concentrated into a smaller frequency region. Each zero corresponds to a degree of freedom and with more degrees of freedom we are able to shape the frequency responses of the filter more precisely and reduce filter errors.

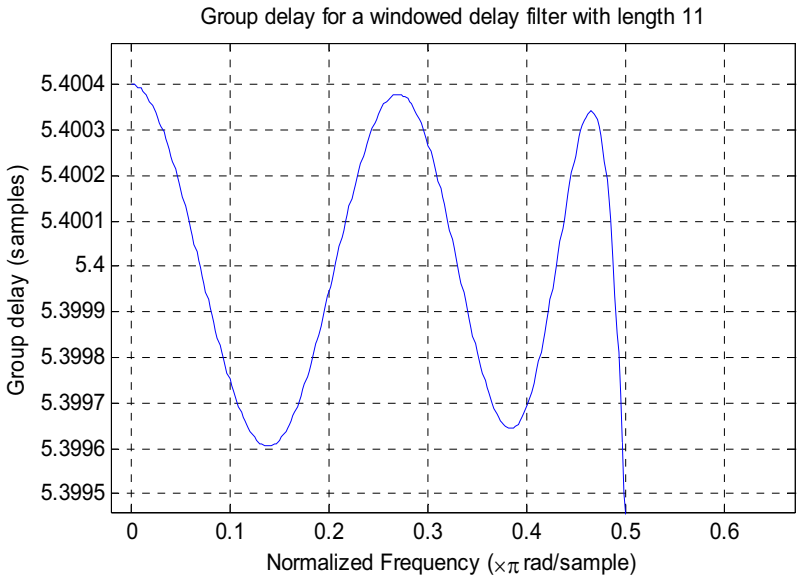


Figure 13: Group delay plot for a delay filter of length 11 that was designed to have a delay of 5.4. The usable bandwidth region for this filter is 0 to 0.5 (normalized frequency) and the peak delay error is 0.0004.

To demonstrate the fractional sampling delay with real hardware we have set up two delay experiments. The first demonstrates the case when the delay is comparable to the sampling period. The second demonstrates fractional sample delays when the delay amount is very small compared to the sampling period. The experimental setup for the first case is similar to the illustration shown in Figure 14. In this experiment we intentionally added a severe phase mismatch to the system providing a delay amount comparable to the sampling period, which can be used to correct for the phase mismatch. A phase difference of 37.28 degrees was added to the two channels by using two SMA cables of different length to connect the outputs of the power divider to the input of the ADCs. The sampling rate of the ADC is 100 MHz at 14 bits and the frequency of the input signals to the ADCs is 34 MHz. Figure 15 shows the time plot of two ADC channels. Because the two waveforms are offset by 37.28 degree, they are completely misaligned.

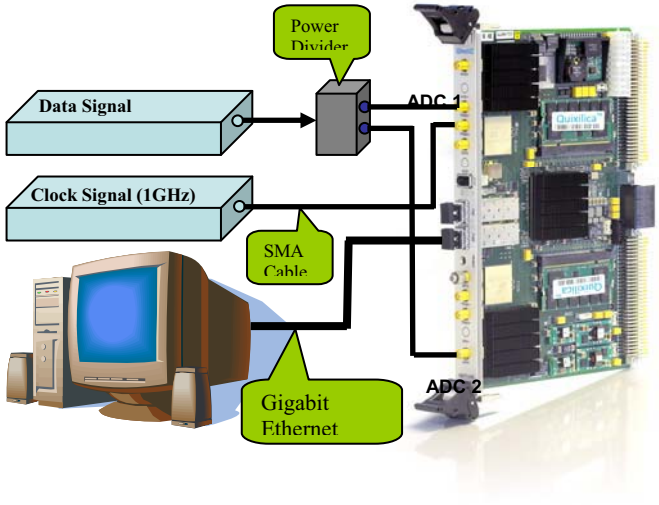


Figure 14: Hardware setup for the delay experiment. The two SMA cables that connect the outputs of the power divider to the input of the ADC have the different lengths.

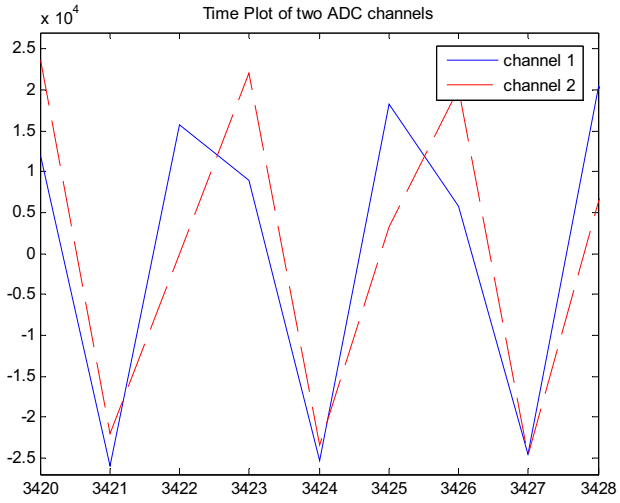


Figure 15: Time waveforms from the two ADCs, sampled synchronously at 100 MHz. The severe misalignment between the two waveforms was caused by the use of cables with different lengths that connect the outputs of the power divider to the inputs of the two ADC channels.

Using a windowed fractional delay filter, a time delay of 3.0459 ns, which is equal to 0.3045 of the sampling period, was applied to the second channel. Figure 16 shows time plots of the data before and after time delay. After implementing the time delay, channel 2, shown in dashed black, is almost identical to channel 1, shown in solid blue. Not

shown here, in the second experiment the delay amount is about 0.0679T and using digital delay filter we were able to compensate for the small mismatches between the two ADC channels.

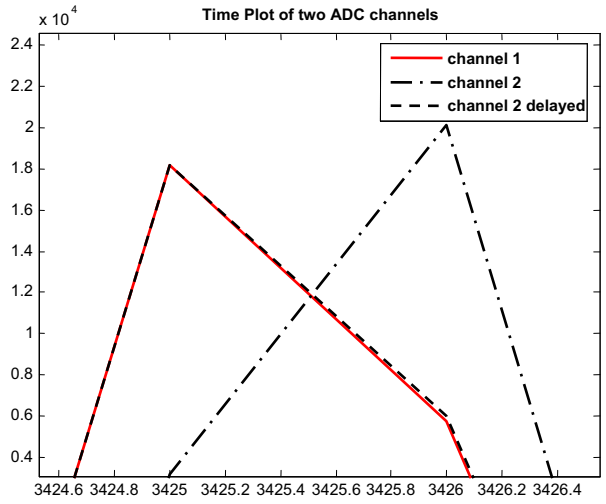


Figure 16: Time waveform of the data from channel 1, channel 2, and channel 2 after calibration (time delay). After digital delay, the time waveform of channel 2 looks almost identical to the time waveform of channel 1.

5.2 Narrowband Adaptive Beamforming

In addition to forming beams, some applications require the steering of the array pattern nulls to mitigate the affects of interference sources. There are numerous adaptive beamforming techniques in the literature that steer nulls toward interferences at the same time while keeping the beam at the desired angle. Most of these adaptive techniques use the narrow band assumption, which greatly simplifies the formulation and solving of the adaptive beamforming problem.

Under the narrow band assumption, the received signals of the array can be modeled as

$$\mathbf{x}(t) = A\mathbf{s}(t) + \mathbf{n}(t), \tag{5}$$

where,

$A = [\mathbf{a}(\theta_1), \mathbf{a}(\theta_2), ..., \mathbf{a}(\theta_K)]$, and $\mathbf{a}(\theta_k)$ is the steering vector for the kth source.

The received power signal for the set of weights $\mathbf{w}$ is

$$\mathbf{w}^H R\mathbf{w}\,, \tag{6}$$

$$R = E[\mathbf{x}(t)\mathbf{x}^H(t)]$$

One of the earliest adaptive beamforming techniques is based on the idea that the optimum weights can be obtained by solving the following constrained equation for w

$$\text{minimize } \mathbf{w}^H R\mathbf{w} \text{ subject to } \mathbf{w}^H\mathbf{v} = \text{constant}\,, \tag{7}$$

where $\mathbf{v}$ is the steering vector of the desired direction. The solution of (6) is given as

$$\mathbf{w} = \alpha R^{-1}\mathbf{v}\,, \quad \text{where } \alpha \text{ is a constant.} \tag{8}$$

Although the weights (8) minimize the output power (6) while pointing the beam to the direction specified by v, it has the effect of steering nulls toward interference sources. The difficulty in solving (8) gives rise to many adaptive beamforming algorithms, which address different practical issues including not having enough samples to accurately estimate R, numerical stability, and algorithm complexity.

Inverting R is a computationally intensive step and requires $O(N^3)$ operations where N is the number of adaptive channels. Other adaptive algorithms may not require the direct inversion of R but their complexity also is directly related to N. Clearly as N increases, the algorithm complexity growths exponentially. In the ideal case, one would want to use all of the channels in the array in the adaptive processing step. To reduce computational requirements in large, wideband practical arrays, many more degrees of freedom are required to suppresses interferences. N is chosen to be the minimum value that enables the adaptive processor to achieve a specified performance in a given environment.

5.3 Subarraying

The classic approach to reduce the number of degrees of freedom and system cost in large arrays is subarraying. In the subarray approach, the array elements are partitioned into M groups and non-adaptive beamforming is used within each group. To reduce the number of expensive receivers in the system the non-adaptive beamforming is usually done in the analog domain. The outputs of each subarray are sampled and then sent to an adaptive processing unit for further processing. Using analog subarraying, one can

reduce the degrees of freedom and number of digital receivers of a system by a factor of N/M, where N is the total number of elements in the array and M is the number of subarrays. Subarraying is an integrated part of large practical wideband array and without subarrays the system may not be technologically or financially feasible. Although some subarraying schemes will reduce system cost and computational requirements, the subarray grouping must be carefully chosen in such a way as not to cause the quantization lobe to rise. Some proposed approaches to reduce quantization lobes include overlapped subarrays, use of subarrays of different optimally chosen sizes, and the related polyomino scheme [26].

The other major aspect of DBF affected by subarraying is the generation of multiple independent beams, which is one of the highly desired features of digital beamforming systems. In the fully adaptive process where we have access to data at every element, one can generate an arbitrary number of simultaneous beams pointing to any desired set of independent angles. In almost all of the existing subarray architectures the number of independent beams is bounded by the shape and gain of the subarray pattern. The use of Subarrays not only affects computational complexity and system cost but it also greatly influences the performance of the adaptive processor. In large, wideband arrays, the subarray architecture is tightly integrated with many other aspects of the system and digital beamforming in particular. The digital beamforming architecture tradeoffs among performance, cost, computation complexity and desired features, such as multiple beams, should not be done without taking into account the system subarray architecture.

5.4 Hardware

The feasibility and limitations of a digital beamforming system is determined by the cost, size, power consumption and processing capabilities of digital hardware. The general purpose processing unit such as an FPGA is often bulky and expensive but it is readily available, fully reconfigurable, and can be integrated into a system very quickly. Highly customized solutions, such as ASICs, are inexpensive to mass produce, they are small, and power efficient, but they require a significant initial investment of time and resources to properly design and test.

The computational resources required in a DBF system are directly related to the system bandwidth, dynamic range and algorithm complexity. Increasing the instantaneous bandwidth will increase the volume of data that must be processed. Increasing the dynamic range will require the arithmetic to have greater precision and algorithms with greater complexity to preserve the dynamic range throughout the computation process. In either case we need to increase the processing resources in order to keep the system response time unchanged. In practice, system bandwidth and dynamic range are fixed by the application so a typical way to reduce the resource requirement is by simplifying the algorithm, which unfortunately degrades system performance. One of the challenges in the field of digital beamforming is that currently desired system bandwidth and dynamic range performance exceeds the capability of present COTS hardware, given practical

limits on budget, size, and power. In these cases tradeoffs must be made between lowering system performances and using highly simplified processing algorithms in order to make the system feasible.

Key pieces of hardware in a DBF system are ADC, FPGA or ASIC, embedded processors and high speed buses to transfer data from the ADC to FPGA and then to other embedded processors. Computation tasks in a DBF system are typically assigned to either FPGA or embedded processors depending on whether or not the task is suitable for parallel computations. Tasks that are highly parallelizable such as the FFT and FIR filtering are usually done at the FPGA, which can apply these algorithms to the incoming data at a much higher rate than embedded processor. Complex serial algorithms not suitable for implementation in the FPGA architecture are assigned to fast processors.

In wideband systems, data must be sent between FPGAs, and from an FPGA to a processor. Ideally, we would integrate the ADC, FPGA and processors into one board and transfer massive amount of data among these components using one of the many standard interfaces. However, in practice we need to send data in one FPGA to a different FPGA or to a processor that resides on a different board at a significant distance away. In this case, a high speed bus that is able to carry data over a significant distance is required. Implementing this complex bus adds computation complexity and cost to the system since the sending and receiving protocols need to be installed in both the sending and the receiving boards.

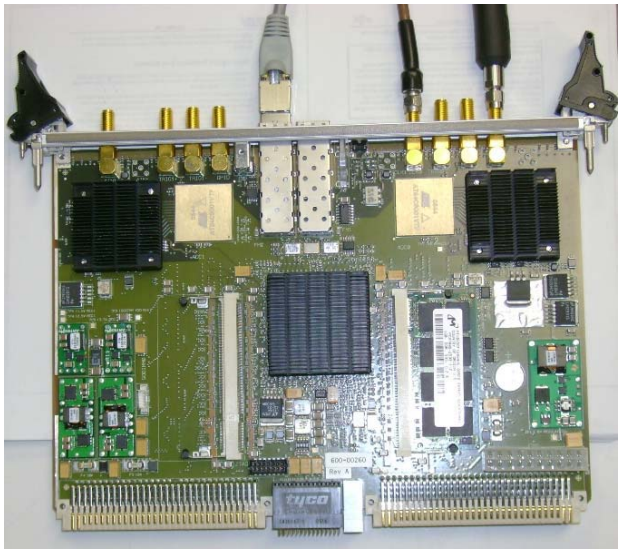


Figure 17: Neptune 2 VSX board from Quixilica with dual 2.2 GSPS ADCs and a Xilinx Virtex II Pro FPGA.

Currently, the highest speed commercial ADC is made by National Semiconductor that can sample up to 3 GSPS at 8 bits and consumes 1.8 watts. The latest FPGA from Xilinx is the Virtex 5. It has twenty four 3.2 Gbps Rocket transceivers for high speed serial communication and a processing capability of 352 GMACS. Figure 17 shows a somewhat obsolete board with high speed ADCs and fairly large FPGA which we have used at AFRL/SNHA for various experiments. Although this board does not have an integrated processor, to apply complex algorithms to the incoming data they must be streamed to a separate processor module via gigabit Ethernet module. The bandwidth limitation in this case is the gigabit Ethernet bus which has a bandwidth of about 90 Mbytes per second.

6. Future Trends

Future trends in DBF that are likely to be realized over the next few years include advances in modularity, multi-functionality, nonlinear digital filtering, receiver-on-a-chip (ROC) technology, wideband nulling, localized signal generation, and use of orthogonal waveforms. These are possible due to the ever advancing state of the art in digital signal processing hardware. Other, longer-term future trends may include the use of switch fabrics as a foundation for adaptive channel hardware that changes configuration in order to cope with changes in mission key performance parameters. Such new hardware may find use in conformal apertures that adhere to nearly every inch of the leading edges and undersides of an aircraft, and be used to adapt the array aperture shape, location, or orientation, in order to optimize array performance or make up for failed portions of the aperture.

6.1 Modularity

Component cost is a key driver in commercial electronics, and it is always an issue in phased array antennas, where RF electronics are notoriously expensive and economy of scale has never yet been achieved, at least not to the degree seen in the automobile industry. However, as DBF technology matures, attention is also being turned to building digital array channel hardware in repeatable blocks that can be concatenated together to form arrays of varying sizes. Scalability of an array architecture is often the intended result of modularity, but with it may come a certain amount of component cost savings.

Our team worked with Applied Radar, Inc., over a four year period to develop a prototype 32-channel, X-band, DBF receive array using COTS components [19, 27]. While cost savings was not a main goal of this project, modularity was, and with it came re-use of components which reduces cost. This hardware contained 2” x 2” x 1” tall custom receiver and transmitter modules, which plugged directly into a custom array processor backplane. The processor backplane facilitated calibration of the array and performed digital beamforming on both transmit and receive. These “plug and play array” modules contained two miniature circuit boards, constituting the Analog Front-End and the Digital Back-End, respectively. As mission needs change, theoretically, one

could swap out the X-band Analog Front-End for one designed at a different center frequency and still utilize the same Digital Back-End, provided the instantaneous bandwidth and the intermediate frequency output from the Analog Front-End was constant. This approach was intended to reduce costs if we chose to demonstrate DBF at a variety of microwave frequencies.

This experimental prototype demonstrated some elementary forms of modularity, but greater degrees of hardware re-use and array scalability will likely be realized as more and more low-cost processor components are pushed into the array aperture. As processor and memory costs drop, less of a premium will be placed on the digital array channel electronics, enabling the channel hardware to be over-designed. This will enable built-in flexibility which will lead to a more generalized channel hardware design, or family of designs, that can be adapted to a variety of mission requirements. This in turn may lead to the use of a common hardware channel module that can be programmed to handle a variety of applications.

6.2 Multi-functionality

A modular common hardware channel also will enable multi-functionality, in which a digital array may be designed to serve more than one mission, sometimes simultaneously. Most aircraft must carry several antennas, and it is common to seek more functionality using fewer antennas. In military aircraft, it is useful to consider an array that could perform both MTI and SAR radar functions. This, however, requires digital array hardware that can be narrowband with high resolution for MTI, but also be wideband with less resolution for SAR. This requires the Analog Front-End electronic circuits to be reconfigurable in order to cope with competing key performance parameters. Again, if low cost processor components are used in the digital array channel, hardware will be at less of a premium and intentional over-designing will result in flexibility and programmability. However, if an array must produce multiple simultaneous beams at different frequencies, or operate in a full duplex communications mode, a greater challenge exists in reducing high levels of interference between the transmit and receive channels, and preventing frequencies that are harmonically related from coupling into the channels from the array apertures. To achieve reasonably good performance, very low loss, high isolation Analog Front-Ends must be designed, coupled with high roll-off bandbass digital filters, and potentially other forms of diversity like orthogonal waveforms.

6.3 Nonlinear Digital Filters

Currently available analog electronics, such as low noise amplifiers and analog filters, and most A/Ds, suffer from inter-modulation products and spurious signals that degrade performance. Recent work in nonlinear digital filters is gaining back some of this lost performance. First, the designer must interrogate the digital array channel hardware with a large set of two and three tone sinusoidal signals spanning the pass band of the

component and measure the resulting spectrum of inter-modulation products. Based on this result, a nonlinear filter is designed that counteracts the nonlinear behavior of the component which was revealed from the measured data set. When the nonlinear digital filter is inserted into the channel hardware path, spur-free dynamic range is increased, and in the case of A/Ds, the effective number of bits is increased, often approaching the actual number of bits in the A/D converter. All of this is done just once per channel, during the initial calibration of the digital array and equalization of its channels, when the array is first built. A dedicated built-in test capability and a dedicated processor would be required to realize dynamic nonlinear filter implementation on the fly, but it is not clear that it would be warranted unless the channel hardware was exposed to great swings in temperature such as on a spacecraft.

6.4 Receiver On a Chip (ROC)

The advent of receiver on a chip (ROC) technology offers great promise for miniaturizing digital channel hardware and pushing digital arrays to higher GHz frequencies. Mixed signal integrated circuit technology, which combines analog and digital circuits on a single chip, is key to the rapid progress in this area. Several companies are engaged in this work and one in-house team in AFRL/SND is developing ROC technology [28]. The designs are generally less complicated than the Digital Tx/Rx Module shown in Figure 3, but the designs realized to date are amazing in their functionality and small size. This technology is an enabler for low cost digital arrays, and may very well be the key to proliferating DBF across many military applications and into the commercial sector.

6.5 Wideband Beamforming and Nulling

Wideband null forming, or nulling, poses a challenge to the very foundation of adaptive beam-forming and direction finding algorithms. Most algorithms were developed based on a narrow band assumption pertaining to the statistics of the array environment [21]. This assumption does not hold when array instantaneous bandwidths go into the hundreds of MHz, and ultimately, the algorithms still produce weights, but the results are often less accurate than desired, if not invalid. One approach to counteract this problem is digital sub-banding, which uses polyphase filters, such as quadrature mirror filters, to split the instantaneous bandwidth into many contiguous narrow sub-bands. Each sub-band is then processed using the traditional algorithms, and the resulting beams and/or nulls can be reassembled, if desired, using polyphase filters.

If the beamforming is deterministic, an alternative method for wideband beamforming is to implement digital time delays using fractional sample delay filters for fine resolution down to fractions of a wavelength, and use shift registers and other approaches for course resolution delays of more than one clock pulse. This approach is not effective for adaptively generated beams or nulls because the computed weights are derived from narrowband algorithms.

6.6 Localized Signal Generation

In digital arrays that require mixers and local oscillator signals, a great penalty is incurred from the additional size and weight of microwave transmission lines at each channel. Ideally, it would be better to generate the LO signal right at the channel and merely route bias voltages and digital command words that control the LO generation. The Digital Tx/Rx Module shown in Figure 3, provides the hardware capability to support local signal generation of both the message carrying signal and all LO signals needed in the Analog Front-End for up-conversion to the final transmit waveform. The LOs are not depicted in Figure 3, but they pass from the DDS WFG to the Analog Front-End Module, and are used in both the Tx and Rx sides of the Analog Front-End Module.

The key challenge in localized signal generation is developing and maintaining phase lock on all LOs across the array when there is apparently no feedback mechanism available for phase locking to a single master oscillator. One way to test the waveforms is to couple a small fraction of the signal power into the receive channel but this could only be done in cases where full duplex operation is not required. Else, one could provide an auxiliary receiver, which drives up the cost of the channel hardware. The goal is to synchronize the formation of the waveforms across the array using the Tx Sync Pulse at each channel. The distribution of synchronizing pulses must account for path length through the processor chain and make corrections for errors at each channel. In our plug and play array channel modules, Applied Radar included clock skewing integrated circuits commonly used in digital sampling oscilloscopes to provide course time delay adjustment to within one clock pulse [27]. Additional time delay resolution can be achieved by imparting fractional sample delays in the beamforming weights or in the channel calibration weights, to counteract errors in sync pulse distribution.

6.7 Orthogonal Waveforms

An orthogonal waveform is typically a digital signal that is part of a set of related digital signals which are constructed according to a design rule that makes them mutually orthogonal in energy over a period of time. Orthogonal waveforms are used in modern communications systems like code-division multiple access (CMDA) cell phones, because they offer added diversity, enabling multiple users to be in the same cell on the same frequency at the same time. In the Digital Tx/Rx Module depicted in Figure 3, the DDS WFG and the Digital Receiver provide the hardware capability to support orthogonal waveforms. While not explicitly shown in the figure, a matched filter can be implemented in the Digital Receiver to sort out multiple orthogonal waveforms. Sorted orthogonal waveforms are then passed to the Receive Waveform Memory, where they wait to be sent through the Channel Processor to the DBF Processor for beamforming. At the DBF Processor, beamforming may be done separately on each orthogonal waveform, resulting in separate beams which may point in the same direction or in different directions.

While greatly used in digital communications systems, orthogonal waveforms have not been in the main stream of radar system design. Much research is being conducted in orthogonal waveforms for use in a variety of applications [29, 30]. One potential benefit may lie in simultaneously transmitting two or more radar pulse sequences at the same frequency, but at different pulse repetition frequencies and embedded in orthogonal waveforms. The beamforming and radar processing would need to handle the orthogonal waveforms, but a single beam could be used to resolve range ambiguities in a shorter time. Orthogonal waveforms may also be used if an array could support multi-functionality, enabling the MTI and SAR functions to occur simultaneously.

7. Conclusion

Our intent for this paper was to provide an update on the current state of the art of digital beamforming technology. We began with a generalized digital array architecture that emphasized the use of distributed processing to reduce the overall data volume and the associated processing load. In describing the roles of the functional blocks and data parameters used in this architecture, we also described some of the main issues of digital array operation, including signal generation, beamforming and calibration. We then showed variations of the key elements of digital array architectures, including the receive channel hardware, the transmit channel hardware, and the processor hardware. For each of these elements, we described the various hardware components which currently form the building blocks of digital array architectures. We delved into various implementation techniques and illustrated the major design trade issues that one must consider in order to maximize performance while minimizing cost and SWaP.

As it stands today, digital beamforming is now a rapidly emerging technology that can be used in specific kinds of real-world applications. Namely, DBF can be used in systems requiring narrow to moderate bandwidths up to few hundred MHz, at frequencies up to X-band, in arrays up to a few hundred elements. At X-band and higher, the digital array channel hardware is not yet miniature enough fit behind a single element, so some form of sub-banding is required. For bandwidths wider than a few hundred MHz, processing load and component cost will likely be the limiting factors. For arrays larger than a few hundred elements the same will be true. Specific cases exist where DBF is absolutely a viable design option and should be considered over conventional analog beamforming. For example, in an airborne UHF array where SWaP is very important, a DBF system employing direct sampling may be used. Such an array will provide considerable savings in weight due to the elimination of the large UHF feed network and cables common in analog phased arrays at this frequency.

In the future, array design engineers should expect to see a wider range of lower cost COTS components, with highly integrated functionality, including receivers on a chip, at frequencies well beyond X-band. Nonlinear filtering will likely be incorporated directly into ADCs and ROCs. Processor and memory chips are expected to continue to improve, and the conversion between analog and digital domains will be achievable at lower cost

and for wider instantaneous bandwidths. As this happens, one would imagine that the cost of digital arrays may drop to levels significantly less than that of conventional phased array antennas. The array designer can then expect to enjoy significantly more degrees of freedom and significantly higher control precision, and have greater number of design options at one's disposal as component cost is reduced. Digital arrays will likely incorporate many of the future capabilities cited in section 6, of this paper, making digital array antennas extraordinarily useful over a wide range of applications. The use of digital arrays and digital beamforming, will likely see widespread use the military, commercial, and scientific sectors.

8. References

[1] Barton, P., "Digital Beamforming for Radar", *Proceedings of the IEEE*, vol. 127, Pt-F 4, Aug, 1980.

[2] Steyskal, H., Rose, J. P., "Digital Beamforming for Radar Systems", *Microwave Journal*, vol. 32, pp. 121-136, Jan, 1989.

[3] Steyskal, H., "Digital Beamforming", *European Microwave Conference*, pp. 49-57, Oct, 1988.

[4] Applebaum, S.P., "Adaptive Arrays", *IEEE Transactions on Antennas and Propagation*, vol. 24, pp. 585-598, Sept, 1976.

[5] Widrow, B., *et al*, "Adaptive Antenna Systems" *Proceedings of the IEEE*, vol. 55, pp. 2143-2159, Dec, 1967.

[6] Steyskal, H, "Digital Beamforming at Rome Laboratory", *Microwave Journal*, vol. 39, No. 2, 1 Feb, 1996.

[7] Herd, J., "Experimental Results from a Self-Calibrating Digital Beamforming Array", *IEEE AP Society International Symposium*, Dallas, TX, 7-11 May, 1990.

[8] Herd, J.S., "Array Element Pattern Correction in a Digitally Beamforming Array: An Experiment Study", *URSI National Radio Science Meeting*, Canada, 1985.

[9] Mailloux, R.J., "Array Failure Correction with a Digitally Beamformed Array", *IEEE Transactions on Antennas and Propagation*, vol. 44, pp. 1543-1550, Dec, 1996.

[10] Pettersson, L., *et al*, "An Experimental S-Band Digital Beamforming Antenna", *IEEE International Symposium on Phased Array Systems and Technology*, Boston, MA, 15-18 October, 1996.

[11] Miyauchi, M., "Development of DBF Radars", *IEEE International Symposium on Phased Array Systems and Technology*, Boston, MA, 15-18 October, 1996.

[12] Garrod, A., "Digital Modules for Phased Array Radar", *IEEE International Symposium on Phased Array Systems and Technology*, Boston, MA, 15-18 October, 1996.

[13] Walden, R.H., "Analog-to-Digital Converter Survey and Analysis", *IEEE Journal on Selected Areas in Communications*, vol. 17, April, 1999.

[14] Bucci, O.M., et al, "Intersection Approach to Array Pattern Synthesis", *IEE Proceedings-Pt. H*, vol. 137, pp. 349-357, Dec, 1990.

[15] Curtis, D., et al, "Conformal Array Control Using Digital Beamforming", *Antenna Application Symposium*, Monticello, IL, Oct 2001.

[16] Couch, Leon W., "Digital and Analog Communication Systems", Prentice Hall, Upper Saddle River, NJ, 2001.

[17] Rabiner, L., and Gold, B., "Theory and Application of Digital Signal Processing", Prentice Hall, Inc., Englewood Cliffs, NJ 1975.

[18] Oppenheim, A., Schafer, R., and Buck, J., "Discrete-Time Signal Processing", Prentice Hall, Inc., Upper Saddle River, NJ, 1999.

[19] Spendley, *et al*, "Initial Demonstration of an X-Band Digital Beamforming (DBF) Receive Array", *2006 IEEE Aerospace Conference*, Big Sky, MT, March 4-11, 2006.

[20] Mallioux, Robert J., "Phased Array Antenna Handbook", Artech House Inc, Norwood, MA, 1994.

[21] Van Trees, Harry L., "Optimum Array Processing", John Wiley & Sons, Inc., New York, 2002.

[22] Gershman, A. B., "Robust Adaptive Beamforming: An Overview of Recent Trends and Advances in the Field", *International Conference on Antenna Theory and Techniques*, 9-12 September 2003, Sevastopol, Ukraine.

[23] "A Technical Tutorial on Digital Signal Synthesis", An Analog Devices Tutorial, Analog Devices, Inc., 1999. www.analog.com

[24] T. I. Laakso, V. Valimaki, M. Karajalainen, and U. K. Laine, "Splitting the Unit Delay", *IEEE Signal Processing Magazine*, Vol. 13, Issue 1, pp. 30-60, Jan. 1996.

[25] T. I. Laakso, T. Saramaki, and G. D. Cain. "Asymmetric Dolph-Chebyshev and Transitional Windows for Fractional Delay FIR Filter Design", *Midwest Symposium on Circuits and Systems*, Rio de Janeiro, Brazil, 13 August, 1995.

[26] Mailloux, R.J., Santarelli, S.G., Roberts, T.M., "Polyomino Shaped Subarrays for Limited Field of View and Time Delay Control of Planar Arrays", *Antenna Application Symposium,* Monticello, IL, 21 Sept 2005.

[27] Curtis, D., *et al*, "32-Channel X-Band Digital Beamforming Plug-and-Play Receive Array", *IEEE International Symposium on Phased Array Systems and Technology*, Boston, MA, 14-17 Oct, 2003.

[28] Quach, T., Wyatt, P., X-Band Receiver Front-End Components in Silicon Technology", *GOMACTech Conference*, Lake Buena Vista, FL, 19-22 March 2007.

[29] Robey, F., *et al*, "MIMO Radar Theory and Experimental Results", *37th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, 7-10 Nov, 2004.

[30] Donnet, B., Longstaff, I. "MIMO Radar, Techniques and Opportunities", *Proceedings of the 3rd European Radar Conference*, September 2006, Manchester, UK.